

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 08-016356

(43)Date of publication of application : 19.01.1996

(51)Int.Cl.

G06F 3/14

G06F 3/14

G06F 12/00

(21)Application number : 07-148857

(71)Applicant : INTERNATL BUSINESS MACH CORP <IBM>

(22)Date of filing : 15.06.1995

(72)Inventor : LI SHIH-GONG
SHRADER THEODORE JACK LONDON

(30)Priority

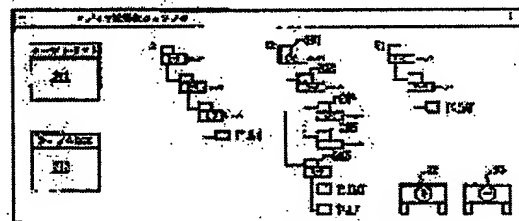
Priority number : 94 265908 Priority date : 27.06.1994 Priority country : US

(54) USER INTERFACE FOR MANAGING MEDIA EXTENSION OF DATA BASE TABLE AND COMPUTER SYSTEM

(57)Abstract:

PURPOSE: To provide an inclusive and topographic map for all the table parts of a data base extending in multiplex virtual or existing storage media.

CONSTITUTION: An addition table icon 22 and a deletion table icon 23 are used for selecting an operation on tree structure. In the tree structure of D:, a directory 202 and a directory 203 are connected to a root directory 201 and directories 204 and 205 show connection to the directory 202. A user can possess performance to view all files belonging to the table of the data base and to enlarge the media of the data base table and he can drag a generated table part entry from one directory position to the different position so as to drop it without being anxious of a media boundary. Since type input to a new directory path (route) is not required by the direct operation method, keyboard input is made small.



LEGAL STATUS

[Date of request for examination] 22.12.1999

[Date of sending the examiner's decision of rejection] 23.03.2004

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

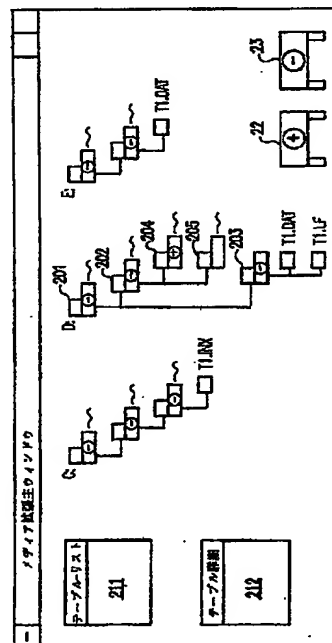
[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

BEST AVAILABLE COPY

(11)特許出願公開番号



【特許請求の範囲】

【請求項1】 コンピュータ・システムにおけるデータベース・テーブルのメディア拡張を管理するユーザ・インタフェースであって、
オブジェクトとカーソルとを表示するためのディスプレイを具備し、マウス・ボタンのシングル・クリックにより前記ディスプレイ上に表示されたオブジェクトを選択するべく、又は前記マウス・ボタンのダブル・クリックによって所与のアクションを呼出すべく、もしくは前記マウス・ボタンの押下保持によるオブジェクトの選択と、選択された該オブジェクトの前記ディスプレイ上における移動と、前記マウス・ボタンの解放とからなるドラッグ・アンド・ドロップ・アクションを呼出すべく使用可能なマウスによって前記オブジェクトと前記カーソルとが制御されるコンピュータ・システム上にて実現され、かつ、データベース中の全てのテーブルのテーブル・リストから構成されるデータ構造を複数の仮想的又は物理的メディアに記憶することにより前記コンピュータ・システム上のデータベース・テーブルのためのメディア拡張を管理するユーザ・インタフェースであって、
前記テーブル・リストを更新しかつ複数のメディアの各々に対する階層的ツリー構造のディレクトリ・ルートを作成することによって前記データベースを初期化するステップと、
前記複数のメディアの各々についてメディア拡張主ウィンドウ内にルート・ディレクトリ・アイコンを表示するステップと、
ユーザによるディレクトリ・アイコン上でのダブル・クリックを検知し、そして前記ディレクトリ・アイコンが、該ディレクトリ・アイコンの下に表示される階層ツリー構造の中に子ディレクトリ・アイコン又はテーブル・パート・アイコンが存在するにも拘わらずこれらを表示していない場合は、表示を拡大することによりこれらの子ディレクトリ・アイコン又はテーブル・パート・アイコンを表示し、一方、存在しない場合は、前記階層ツリー構造の表示を取止めて前記ディレクトリ・アイコンのみを表示するステップと、
ユーザによる前記テーブル・パート・アイコン上でのダブル・クリックを検知し、そして前記ユーザに対して、前記テーブル・パート・アイコンに対応するテーブルのフィールドを変更する入力を行うよう促すステップと、
ユーザによる一方のディレクトリ・アイコンから他方のディレクトリ・アイコンへの前記テーブル・パート・アイコンのドラッグ・アンド・ドロップ処理を検知し、そしてその結果変更された前記階層ツリー構造の表示中のディレクトリ・パスを更新しかつ前記データベースの前記データ構造を更新するステップと、
前記メディア拡張主ウィンドウ中に追加テーブル・パート・アイコンと削除テーブル・パート・アイコンとを表示するステップと、

ユーザによる前記追加テーブル・パート・アイコンの前記ディレクトリ・アイコンへのドラッグ・アンド・ドロップ・アクションを検知し、そして前記ユーザに対して、前記ディレクトリ・アイコンに対応するディレクトリへ追加されるべきテーブルのフィールドを変更する入力を行うよう促すステップと、
ユーザによるディレクトリ・ツリー中の前記テーブル・パート・アイコンの選択とユーザによる前記削除テーブル・パート・アイコンの選択とを検知し、そして選択された前記テーブル・パート・アイコンを前記メディア拡張主ウィンドウの表示から取除き、かつ前記データベースを更新するステップとを含むユーザ・インタフェース。

【請求項2】 前記メディア拡張主ウィンドウ内に、テーブル・リスト副ウィンドウを表示するステップと、
前記メディア拡張主ウィンドウ内に、テーブル詳細リスト副ウィンドウを表示するステップと、
ユーザによる前記テーブル・リスト副ウィンドウからの選択を検知し、そしてテーブル・データ構造を構文解析することによって、テーブル詳細リスト・エントリを用いてテーブル詳細リストを更新するステップと、
ユーザによる前記テーブル詳細リスト副ウィンドウからの選択を検知し、そして前記ユーザによる選択がメイン・テーブル・エントリである場合、前記ディレクトリ・ツリー中の全てのテーブル・パートを強調表示し、一方、前記ユーザによる選択がテーブル・パート・エントリである場合、前記ディレクトリ・ツリー中の個々の前記テーブル・パート・アイコンを強調表示するステップとを含む請求項1に記載のユーザ・インタフェース。

【請求項3】 データベース・テーブルのメディア拡張を管理するコンピュータシステムであって、
システム・バスに接続された中央演算処理ユニットと、
前記システム・バスに接続され、かつユーザにより制御されるマウスと、
前記システム・バスに接続され、かつデータベース中の全てのテーブルのテーブル・リストからなる該データベースのデータ構造を記憶する複数の仮想的又は物理的メディアと、
前記システム・バスに接続され、かつマウス・ボタンのシングル・クリックによりディスプレイ上に表示されたオブジェクトを選択するべく、又は前記マウス・ボタンのダブル・クリックによって所与のアクションを呼出すべく、もしくは前記マウス・ボタンの押下保持によるオブジェクトの選択と、選択された該オブジェクトの前記ディスプレイ上における移動と、前記マウス・ボタンの解放とからなるドラッグ・アンド・ドロップ・アクションを呼出すべく前記ユーザにより使用可能な前記マウスによって制御されるグラフィック・ユーザ・インタフェース中のオブジェクトとカーソルとを表示するディスプレイとを有し、

3

前記中央演算処理ユニットが、前記テーブル・リストを更新しかつ複数のメディアの各々に対する階層的ツリー構造のディレクトリ・ルートを作成することによって前記データ・ベースを初期化するべく、前記複数のメディアの各々についてメディア拡張主ウィンドウ内にルート・ディレクトリ・アイコンを表示するべく、ユーザによるディレクトリ・アイコン上でのダブル・クリックを検知して、前記ディレクトリ・アイコンが該ディレクトリ・アイコンの下に表示される階層ツリー構造の中に子ディレクトリ・アイコン又はテーブル・パート・アイコンが存在するにも拘わらずこれらを表示していない場合は、表示を拡大することにより前記子ディレクトリ・アイコン又は前記テーブル・パート・アイコンを表示し、一方、存在しない場合は、前記階層ツリー構造の表示を取止めて前記ディレクトリ・アイコンのみを表示するべく、ユーザによる前記テーブル・パート・アイコン上でのダブル・クリックを検知して、前記ユーザに対して前記テーブル・パート・アイコンに対応するテーブルのフィールドを変更する入力を行うよう促すべく、ユーザによる一方のディレクトリ・アイコンから他方のディレクトリ・アイコンへの前記テーブル・パート・アイコンのドラッグ・アンド・ドロップ処理を検知して、その結果変更された前記階層ツリー構造の表示中のディレクトリ・パスを更新し、かつ前記データベースの前記データ構造を更新するべく、前記メディア拡張主ウィンドウ中に追加テーブル・パート・アイコンと削除テーブル・パート・アイコンとを表示するべく、ユーザによる前記追加テーブル・パート・アイコンの前記ディレクトリ・アイコンへのドラッグ・アンド・ドロップ・アクションを検知し、そして前記ユーザに対して、前記ディレクトリ・アイコンに対応するディレクトリへ追加されるべきテーブルのフィールドを変更する入力を行うよう促すべく、ユーザによるディレクトリ・ツリー中の前記テーブル・パート・アイコンの選択とユーザによる前記削除テーブル・パート・アイコンの選択とを検知して、選択された前記テーブル・パート・アイコンを前記メディア拡張主ウィンドウの表示から取除き、かつ前記データベースを更新するコンピュータ・システム。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明はコンピュータのグラフィック・ユーザ・インタフェースに関係している。特に、ユーザに包括的に地形学的 (topographical) なデータベース・テーブル・パート (part) の地図を提供するデータベース・テーブルのメディア (media) 拡張のためのインタフェースに関する。

【0002】

【従来の技術】パーソナル・コンピュータは今日、多くの人々によって様々な目的に盛んに使用されている。かつてのメインフレームやミニコンピュータ同様に、パー

4

ソナル・コンピュータの最も重要な応用の一つはデータベース・プログラムである。このデータベース・プログラムは、会計プログラム、個人情報管理、顧客情報や電話帳、そして系図録やその他の事項のベースとなる。先ず、パーソナル・コンピュータのディスク容量には制限があるから、その上に構築されるデータベースは制限を余儀なくされる。更に、データベース・プログラムをセット・アップして使用するためにはあるレベルのプログラム作成の技術を必要とする。このことは多くのユーザに対する足かせになることを意味する。しかし最近になってパーソナル・コンピュータのディスク容量は劇的に大きくなっており、そしてデータベース・プログラムに対するユーザ・インタフェースも明解になり、多数のユーザがデータベースを利用できるようになった。

【0003】ディスク容量の増大には様々な要因がある。先ず非常に小さなパッケージに非常に高密度のハード・ディスクが生産できたことである。この結果現在では500メガバイト (MBs) 以上のハード・ディスクをパーソナル・コンピュータに実装させることが一般的になった。この大きなハード・ディスクは多数のパーティション (区画) に分割され、各パーティションは個別の仮想的ハード・ディスクとして取り扱われる。以前は、さらに容量を増やそうとする場合、個別にかつ物理的にハード・ディスクを追加して第2のハード・ディスクとしてパーソナル・コンピュータをグレードアップしていた。仮想的ハード・ディスク、物理的ハード・ディスクあるいはその組み合わせのいずれの場合においてもハードディスクは "C:"、"D:"、"E:" 等のドライブとして区別される。一時期、アップル (Apple) コンピュータの標準であったスモール・コンピュータ・システム・インタフェース (SCSI: スカジ, Small Computer system Interface) は、現在、パーソナル・コンピュータの標準になっている。スカジ・インタフェースによって幾つかのハード・ディスクを環状連鎖 (daisy chain) することができるのでコンピュータの記憶容量を容易に拡大可能である。

【0004】今日一般的に実現されているユーザ・インタフェースはいわゆるグラフィック・ユーザ・インタフェース (GUI: Graphical User Interfaces) である。GUIにおいてデータベースは階層的ツリー (木) 構造のファイル・アイコンとして表現され、マウス制御されるカーソルを用いてファイルを選択して操作できる。非常に高度なレベルのプログラム技術を要求していた過去のオペレーションは、現在、プログラム作成経験のない者であっても、単にファイルを選択し、オペレーションを選択し、アイコンをドラッグして所定の場所に持って行く (ドロップ) ことによって、そしてこの種のインタフェースで一般的に採用されている類似の機能を使うことによって実行される。

【0005】ディスク容量の拡大と現今のデータベース

50

5

・プログラムの扱い易さの結果、多数の人々は大きなデータベースを作るようになり、その中にはハード・ディスクやパーティションの容量をも越えるものも現れるようになった。これらのデータベースのデータにアクセスするためのデータベースのテーブルは大きくなり、容易にアクセスすることが難しくなった。テーブルのデータ量が大きくなるにつれ、その中の実際のデータとそれらをサポートする特性の双方について、データベースのテーブルをドライブのパーティションの異なるディレクトリに、あるいは異なるメディア(media)のドライブにまたがって分割する機能に対する要求が生まれた。しかしながら、データベースがこのような形でテーブルを分割する機能を持つならば、ユーザはさらに、テーブルが何処にあるかを知る地形学的(topographical)地図を必要とし、そしてテーブルをディレクトリに亘って分割し続けるための簡単な方法あるいはテーブルをより小さい部分とより大きい部分に統合し続ける方法を必要とする。このように即席で地図を提供し、図形的なフロント・エンドを利用するインタフェースはデータベース管理者の生産性を高めるものである。

【0006】

【発明が解決しようとする課題】それ故に本発明の一つの目的は、パーソナル・コンピュータやワークステーション等のコンピュータにおけるデータベース・テーブルのメディア(媒体)拡張を管理するためのグラフィック・ユーザ・インタフェースを提供することである。本発明のもう一つの目的は、一つのテーブルに属する全てのファイルを素早くそして一義的に識別できるように一つのデータベースの全てのテーブル・パート(table parts)の、包括的且つ地形学的地図を与えるものである。

【0007】

【課題を解決するための手段】本発明によれば、データベース管理者のようなユーザが、作成されたテーブル・パートを一つのディレクトリからドラッグし、それを別の場所に持って行くこと(ドロップ)ができるようなデータベース・テーブルのメディア拡張のためのユーザ・インタフェースを提供する。この直接処理によって、テーブル・パートの位置を動かしたり、あるいは新規テーブル・パートを作成するために必要な新しいディレクトリの経路(パス)をタイプ入力する必要がなくなるから、キーボードの入力は、最少になる。このユーザ・インタフェースは、データベース中の全てのテーブルが位置する場所をグラフィック的に見るためにデータベース管理者によって用いられる。

【0008】データベース・テーブルのメディア拡張に対するユーザ・インタフェースは、ユーザ(例:データベース管理者)の考え、あるいは見通しに合致するモデルに基づくユーザ・インタフェースを提供するから一義的である。テーブルに属する全てのファイル(あるいはデータベースの各部分)が素早くそして一義的に識別さ

6

れるような態様で、データベースの全てのテーブル・パートの地図をわかりやすく且つ地形学的にユーザに対して提供する。直接操作の手法が広範囲に用いられ、キーボード入力を最少とすることによって使用の容易さを促進する。例えば、データベース管理者は作成されたテーブル・パート・エン트리(table part entry)を一つのディレクトリ位置からドラッグし、別の所に置く(ドロップ)。新しいディレクトリ経路(パス)をタイプする必要がない。性能上の問題点を発見しやすくするために、データベース管理者は、データベース中の全てのテーブルが置かれている場所をグラフィック的に見るためにデータベース・テーブルのメディア拡張を利用することができる。

【0009】

【実施例】図1には本発明を実現できる代表的ハードウェア環境が示されている。このハードウェアはInternational Business Machines(IBM)社のPS/2パーソナル・コンピュータのようなパーソナル・コンピュータあるいはIBM社のRS/6000ワークステーションのようなワークステーションであってもよい。ハードウェアは中央処理装置(CPU)10を含む。このCPUはインテル社の386、486あるいはペンティアム(Pentium)・マイクロプロセッサのような複合命令セット・コンピュータ(CISC:complex instruction set computer)・マイクロプロセッサあるいはIBM社のPowerPCマイクロプロセッサのような縮小命令セット・コンピュータ(RISC:reduced instruction set computer)・マイクロプロセッサであってもよい。このCPUはシステム・バス12に接続されており、さらにこのバス12にはランダム・アクセス・メモリ(RAM)14、リード・オンリ・メモリ(ROM)16、入出力(I/O)アダプタ18そしてユーザ・インタフェース・アダプタ2が接続されている。RAM14はアプリケーション・プログラムとデータの一時的な記憶に用いられ、ROM16の方は通常、基本入出力システム(BIOS)・コードを含む。I/Oアダプタ18は一つ以上の直接アクセス記憶装置(DASD)に接続されるが、この実施例では一つのディスク・ドライブ20として表されている。このディスク・ドライブ20は通常コンピュータのオペレーティング・システム(OS)やいろいろなアプリケーション・プログラムを格納し、これらのプログラムはシステム・バス12を通してRAM14に選択的にロードされる。I/Oアダプタ18は、例えば集積デバイス・エレクトロニクス(IDE:Integrated Device Electronics)・インタフェース規格あるいはスカジ(SCSI)インタフェース規格を持つ。IDEの場合には、I/Oアダプタ18は通常、ドライブ"C:"、"D:"と符号化して二つのディスク・ドライブを並列にもつ。SCSIの場合には、I/Oアダプタは環状連鎖に接続した9台のディスク・ドライブまで保有できる。ユーザ・インタフェース・アダプタ2にはキーボード4、マウス6、マイクロホン32、そして/あるいは他の装置(例えば図

50

示されていないがタッチ・スクリーン等)が接続されている。タッチ・スクリーンの場合はオプションとしてディスプレイ38に設置される。ここではディスプレイを陰極線管ディスプレイ(CRT)として表しているが、液晶ディスプレイ(LCD)であってもよい。ディスプレイ38はディスプレイ・アダプタ34によってシステム・バス12に接続される。オプションとして通信アダプタ34がバス12と、ローカル・エリア・ネットワーク(例えば、IBMのトークン・リングLAN)のようなネットワークに接続される。あるいは通信アダプタはパーソナル・コンピュータやワークステーションを、広域エリア・ネットワーク(WAN)の役割を成す電話線に接続するためのモデムであってもよい。

【0010】本発明の実施例の以下の記述では、ハード・ドライブ・パーティション(harddrive parttions)に関するテーブルの分割はIBMのパーソナル・コンピュータ用オペレーティング・システムOS/2の下でのデータベースを考えている。IBMのAIXオペレーティング・システムのようなユニックス(Unix)・オペレーティング・システムの下でのデータベースに対しては、テーブルの物理的位置は論理的位置ほどには明解ではない。例えば、OS/2オペレーティング・システムではハード・ドライブをこれらのドライブのディレクトリと一致して"C:"、"D:"として参照するが、AIXオペレーティング・システムでは単にディレクトリに関係しているだけである。テーブルが分割される物理的ドライブはこのレベルでは、無関係である。記述される実装では、いかなる方法でメディア拡張に対するユーザ・インタフェースがOS/2データベースに対して構築されるかを考える。AIXデータベースはディスク・ドライブの参照を許さないから、この設計のサブセットである。更に、この設計は特性、パラメータ、そして実際のデータベースのメディア拡張機能をもった実装が有する類似のものを含むように修正される必要がある。

【0011】図2はパーソナル・コンピュータの表示画面に表示されるメディア拡張主ウィンドウの例を示す。このウィンドウはドライブのツリー構造を示しており、ディスクまたは複数のディスク(仮想的もしくは物理的ディスクまたはこれらの組み合わせ)のツリー構造のディレクトリに従って(例えばツリーのルートから)、ユーザは関係を知ることができる。ルート・アイコンとディレクトリ・アイコンとはフォルダ(folder)の形で表される。ユーザは一つの閉じたフォルダを選択し展開して、全てのサブディレクトリと、選択し、拡張されたテーブルの全てのファイルを見ることができる。あるいは、ユーザは開いたフォルダを選択し、そのサブディレクトリとファイルを単一のフォルダに納めることもできる。この機能はOS/2ファイル・マネージャで与えられているディレクトリのツリー構造に関するものと非常に似ているが、ある重要な差異がある。"C:"として示される

ような各ドライブはそれ自身のドライブ・ツリーを持っている。データベース管理者は主ウィンドウのアクション・バー(action bar)の選択やその他の機能を使ってドライブ・ツリーを追加したり、削除することができる。この機能を使うと、データベース管理者は問題にしているドライブとディレクトリとを個々のユーザ環境に合わせることもできる。例えば、IBMのOS/2のLANサーバ・ネットワーク・マネージャやIBMのAIXのOSにおける分散ファイル・システム(DFS: Distributed FileSystem)では、特定のワークステーションのユーザには関係がないドライブとディレクトリとがシステムに多数存在する。データベース管理者はこれらのドライブとディレクトリとを画面に表示しないようにできる。

【0012】図2の主ウィンドウに別の情報を表示することができる。例えば、データベース管理者(ユーザ)がメディアを旨く使えるように各メディアの物理的容量(総容量と残容量)を表示する。

【0013】図3には図2のメディア拡張主ウィンドウで用いられる数種のアイコンが示されている。これらのアイコンは、基本的に2種類のタイプがある。つまり、テーブル・アイコンとディレクトリ・アイコンである。テーブル・アイコンはテーブル・パート・アイコン21と、追加テーブル・アイコン22と、削除テーブル・アイコン23である。テーブル・パート・アイコンはディレクトリ・アイコンと一緒に用いられて、メディア拡張主ウィンドウ表示のツリー構造を形成する。一方、図2の右下角に示されている追加テーブル・アイコンと削除テーブル・アイコンはツリー構造からテーブルを削除したり、追加したりするようなツリー構造に関する操作の選択に用いられる。ディレクトリ・アイコンは子アイコンをもたないディレクトリ24と、子アイコンをもつがその表示のないディレクトリ25と、子アイコンをもち表示もされているディレクトリ26を含む。ここで"子(children)"とはテーブル・パート・オブジェクトのサブディレクトリである。ここで、特に図2のD:のツリー構造に注目すると、ルート・ディレクトリ201があり、この201にディレクトリ202と203が接続されている。この202と203はアイコン26によって表されている。二つのディレクトリ204と205がディレクトリ202に接続されている。これらのディレクトリ204と205はアイコン25と24でそれぞれ表現される。アイコン21によって表現される二つのテーブルがディレクトリ203に接続されている。このディレクトリ203は"開いて(open)"おり、テーブル・パートを持つ。それ故にこのディレクトリに対するテーブル・アイコンが表示されており、これらのテーブル・アイコンはユーザによって選択される。

【0014】図2のツリー構造に加えてメディア拡張主ウィンドウはさらに一つ以上の副ウィンドウを表示する。例えば、図2のテーブル・リスト・ウィンドウ21

1とテーブル詳細ウィンドウ212が表示される。ユーザがテーブルを処理したいときは、テーブル・リスト・ウィンドウ211(図4に詳細を示す)からテーブルを選ぶ。テーブル・リスト・ウィンドウは、テーブルの大きさと属性と共にデータベースのテーブルのリストを示す。テーブルをリストから選択すると、主ウィンドウのディレクトリ・ツリー構造の全ての該当する「パート(parts)」(例えばファイル・アイコン)が光る(強調表示)。テーブル・パートが表示中のディレクトリにないときは、閉じたディレクトリが展開されるか、あるいは別のディレクトリ・ツリーが加えられる。尚、インタフェースはユーザに対して、選択されるテーブルに属するファイルを含むディレクトリ・フォルダ・アイコンを閉じないようにするかまたは強調表示されたテーブル・パートを含むドライブ・ツリーを除去しないようにすることができる。こうして全てのファイルが画面上に表示されることを保証する。

【0015】前述のように拡張されたテーブルのパートはディレクトリ・ツリーのファイルとして示される。但しこれらのファイルはツリーに示されるファイルだけである。CONFIG.SYSファイルと全ての不適切なファイルは表示されない。テーブル・パート・ファイルはバイナリ・ヘッダかあるいは特別で一義的なファイル名の拡張子によって他のファイルから区別できる。ファイルの名前は、そのファイル・アイコンのすぐ次にあらわれる。例えば、TABLE1.ONE、TABLE1.TWO、TABLE1.DAT、TABLE1.INX等である。主ウィンドウに一つ以上のテーブルのパートを表示できる。カレント・テーブル(選択されたテーブル)は全て、他のテーブルから区別するために、そのファイル名とファイル・アイコンの表示をリバース(反転)にできる。さらに、データベース管理者はあるテーブルの全てのファイル名を他のテーブルと違った色あるいはフォントにすることができるのでユーザはテーブルの配置状況を的確に把握できる。

【0016】あるテーブルが選択されると、テーブル詳細ウィンドウ212(詳細図は図5)は、それが画面表示されていないときは画面にあらわれる。テーブルが既にオープンされているときは、選択されたテーブルがこのリストに加えられるか(新しいテーブル)、または高輝度に光らされる(リストに既に含まれている場合)。このウィンドウはユーザにファイルの名前、ディレクトリ・エントリ、そしてサイズ等の情報を持った特別のテーブルの全てのファイルのクラスタ(集合)・リストを提供する。データベース・テーブルに一義的な他の属性についてもこのリストに加えることができる。ユーザはこのリストを見る事によって主ウィンドウに戻ることもなくテーブルの全てのファイルとそれらの属性を知ることができる。各クラスタの一番上に全てのファイルの総計サイズとテーブルが展開する総計ファイル数が示される。クラスタはテーブル名とそのテーブル・パートのエ

ントリとして定義される。

【0017】図3の追加アイコン22と削除アイコン23が図2にも表示されている。追加(又は作成)パート・アイコンはディレクトリ・アイコンのところまでドラッグされ、そのアイコン上に置かれる(ドロップ)。そして、追加パート・ダイアログ・ボックスがファイル名とそのディレクトリ・パスを伴って表示される。メディア拡張のデータベース作成に対して一義的な他のパラメータもここに表示される。削除パート・アイコンも同じように動作する。テーブル・パートを変更するためにデータベース管理者はテーブル・パートのアイコン上でダブル・クリック(2回のクリック)をする。これにより、チャージ(Charge)・パート・ダイアログ・ボックスが表れる。このダイアログ・ボックスには、ファイルのサイズのような付加的リード・オンリ情報を伴った追加パート・ダイアログ・ボックスに関する全ての情報が含まれている。

【0018】テーブル・パートを作成あるいは削除する直接操作を用いる代わりにデータベース管理者はテーブル・パートに加えたいフォルダを選択し、アクション・バーから追加アクションを選ぶことができる。あるいはデータベース管理者は変更パート・ダイアログ・ボックスの削除ボタンを選択するかあるいは当のアイコンを選択してデリート・キーを押すことができる。さらには、各ファイル・アイコンは第2番目のマウス・ボタンを押下保持することにより起動されるコンテキスト・メニューをもつことができる。このメニューによってデータベース管理者は変更ないし削除パート操作を選べる。ディレクトリ・アイコンは追加パート操作を持ったコンテキスト・メニューをもつ。

【0019】本発明のオペレーションは図6から図9までのフロー図によって最も良く例示される。図6は初期化プロセスを示す。プログラムがスタートすると、機能ブロック501ではテーブル・リストがデータベースの全てのテーブルについて更新される。次に機能ブロック502ではデータベースがオープンされ、それから機能ブロック503ではテーブルの問い合わせリストが生成される。機能ブロック504ではテーブル・パート内部データ構造が配置決めされ、続いてテーブル・リスト・ボックス表示が該データ構造から決められる。プログラムのこの時点で、機能ブロック506のドライブ・ツリーの更新が行われる。機能ブロック507では、ディスク・ドライブの全てのドライブ・パーティションが読み取られる。機能ブロック508では、各ドライブ・パーティションに対して階層にディレクトリ・ルートが作られる。機能ブロック509ではルートの下の第一番目の子ディレクトリがツリー形式で表示される。ここまですが初期化処理であり、機能ブロック510ではメイン(主)・プログラム・ループに入る。プロセスは接続子"B"に導かれて図7(次に参照される)の一番上に行

く。

【0020】図7から図9まではメイン・プログラムの論理が示されている。処理はユーザとの対話形式であり、この処理中にプログラムは、項目を選択するかあるいは操作の呼出またはドラッグとドロップ操作のようなマウス・カーソルを使った他の操作を呼び出すためにマウス・ボタンをシングル・クリック（一回のクリック）するとかダブル・クリックするとかしてユーザが入力した一連の可能な入力信号を検出する。従って図7から図9までのメイン・プログラムのループは可能なユーザ入力を検出して呼び出された処理を行うか、あるいはダイアログ・ボックスを表示してユーザに更にキー入力を促すかである。

【0021】図7の最初は判断ブロック511であり、ここではユーザがテーブル・リストからテーブル・エントリを選択したかどうかを判断する。選択した場合は機能ブロック512において、テーブル・パート・データ構造を解析することによってテーブル詳細リストがテーブル詳細リスト・エントリでリフレッシュされる。その後に接続子“A”を通過して図6のメイン・プログラム・ループ510に戻る。しかし、テーブル・エントリが選択されないときは判断ブロック513でテーブル詳細リストからエントリを選択したかどうかを判断し、選択された場合はさらに判断ブロック514で選択したエントリがメイン・テーブル・エントリであったかどうかを判断する。メイン・テーブル・エントリが選択されている場合は判断ブロック515でディレクトリ・ツリーの全てのパートが高輝度（ハイライト、強調表示）にされてユーザによく見えるようにし、その後に接続子“A”を通過して図6のメイン・プログラム・ループ510に戻る。他方、判断ブロック514でメイン・テーブル・エントリが選択されなかったときは機能ブロック516において、ユーザがテーブル・パート・エントリを選択したとみなされる。この場合には機能ブロック517で個別のテーブル・パート・アイコンがハイライトにされ、その後に接続子“A”を通過して図6のメイン・プログラム・ループ510に戻る。判断ブロック518でユーザがテーブル・パート・アイコンを一つのディレクトリから別のディレクトリに動かした場合は、機能ブロック519で表示の変更ディレクトリ・パス、データ構造、そしてデータベース等全てが更新され、その後に接続子“A”を通過して図6のメイン・プログラム・ループ510に戻る。それ以外の場合は、処理は接続子“C”を通過して図8に行く。

【0022】図8の判断ブロック520ではユーザがディレクトリ・アイコン上でダブル・クリックしたかどうかを判断する。そうであれば、さらに判断ブロック521でダブル・クリックされたエントリはディレクトリかあるいはその下のテーブル・パートアイコンであるかどうかの判断が行われる。もしディレクトリであれば、機

能ブロック522でディレクトリ・エントリは単にディレクトリ・アイコンだけがツリーの中でその子オブジェクト無しと表示されるように表示において縮小される。もしそうでなければ、機能ブロック523においてディレクトリ・エントリは拡張され、その結果としてその子ディレクトリとファイル・オブジェクト（もし有れば）がツリー構造に表示される。いずれの場合もその後に接続子“A”を通過して図6のメイン・プログラム・ループ510に戻る。ユーザがテーブル・パート・アイコンに対してダブル・クリックしたときは判断ブロック524で検知される。そして機能ブロック525において変更／表示テーブル・パート対話ブロックが表示される。次に判断ブロック526において、変更／表示テーブル・パート問合わせが受理されるかどうかに関するキー入力をシステムは待つ。受理されないときは機能ブロック527において全ての変更が無視され、その後に接続子“A”を通過して図6のメイン・プログラム・ループ510に戻る。受理されたときは機能ブロック528において内部データ構造、表示、そして物理的データベースへの変更が記憶され、その後に接続子“A”を通過して図6のメイン・プログラム・ループ510に戻る。判断ブロック529では、ユーザがテーブル・パートを選択し、それを削除するかまたはそれをシュレッダー／ごみ箱の削除アイコンにドラッグするかどうかの判断をする。このときは判断ブロック530において削除処理を行うかどうかの確認が行われる。確認がとれた場合は機能ブロック531において、（1）テーブル詳細リストをリフレッシュし、（2）ディスプレイから該当のテーブル・パート・アイコンを除去し、（3）物理的データベースを更新することによってテーブル・パートはデータベースから除去される。いずれの場合にも、その後に接続子“A”を通過して図6のメイン・プログラム・ループ510に戻る。

【0023】ユーザが追加テーブル・パート・アイコンをドラッグしてディレクトリ・アイコンにもって行く場合は図9の判断ブロック532で検知される。このときは機能ブロック533で追加テーブル問合わせボックスが表示される。次の処理は判断ブロック534において、ユーザが追加テーブル問合わせに対して受理したかどうかの入力待ちとなる。受理された場合は機能ブロック535において、（1）テーブル詳細リストをリフレッシュし、（2）テーブル・パート・アイコンをディスプレイに加え、（3）物理的データベースを更新してデータベースに加える。そうではなく、受理されない場合は機能ブロック536において追加テーブル問合わせの情報が廃棄される。いずれの場合も、その後に接続子“A”を通過して図6のメイン・プログラム・ループ510に戻る。ユーザがこのプログラムから抜け出る場合は判断ブロック537で検知され、機能ブロック538でプログラムは終了する。そうでないときは機能ブロッ

ク539において正しい操作または項目の選択が行われなかったという宣言が出力される。その後接続子“A”を通して図6のメイン・プログラム・ループ510に戻る。

【0024】図6から図9までのフロー図によって例示された処理に適合するデータベースのデータ構造が図10に例示されている。このデータ構造は二種類のテーブルの形をしている。テーブル・データ構造とテーブル・パート構造である。構造1からMまでのリンクされたリストとして示されているテーブル・データ構造は、テーブル名、ファイル数、総計のサイズ、そして最近の更新時間(日と時間)を表したエントリを持つ。テーブルのデータベース設計に一義的な他の属性をこれらの構造のいずれかに含ませうる。テーブル・パート構造は対応するテーブル・データ・テーブルのポインタ・ファイルによって指し示され、そしてファイル名、カテゴリー・タイプ(INX,DAT,その他)、それに最近の更新時間(日と時間)を含む。先と同様に、テーブルのデータベース設計に一義的な他の属性をこれらの構造のいずれかに含ませうる。この例はテーブル・データ構造1に対するテーブル・パート構造1からAまでのリンクされたリストを示す。テーブル・データ構造Mに対する1からBまでのテーブル・パート構造のリンクされたリストもまたある。各テーブル・パート構造は一つ以上のテーブル・パート構造のリンクされたリストを持つ。

【0025】本発明をさらに洗練することが可能である。例えば、データベース管理者は次のオペレーション、移動(Move)、分割(Split)、組み合わせ(Merge)を行うことができる。

【0026】移動オペレーションに対してメディア拡張管理システムは移動するファイルを配置するに十分な容量が対象メディアにあることを検査する。この管理システムはこの検査情報をデータベース管理者に与え、そして必要であれば(例えばダイアログ・ボックスを使って)確認をとる。複数ファイルが選択されたときの移動オペレーションに対して、この容量検査には厳しいものがある。管理システムは、対象メディアが十分な容量を持ち、全ての選択ファイルを同時に移動できなければ移動オペレーションを実行するべきでない。

【0027】分割オペレーションに対してデータベース管理者は一つのファイルを選択して複数のファイルに分割するように選択できる。分割オペレーションを指定する手段はプル・ダウン・メニュー、キーとマウスの組み合わせ、あるいは他の類似の操作を含む。

【0028】組み合わせオペレーションに対してデータベース管理者はまず複数ファイルを選択し、それから組み合わせオペレーションを選択する。システム側は選択された複数のファイルが組み合わせられるかどうかの検査を行う。もし、選択されたファイルが現在同じメディアにないときはシステムは対象メディアの指定を促す。

対象メディアが指定されたときにはシステムは、組み合わせを行うのに足るだけの十分な容量がメディアにあることを検査する。十分な容量がないときはこの情報をユーザに知らせ、組み合わせオペレーションを終了する。容量が十分あるときはシステムはオプションとしてユーザに確認を促すことができる。前と同じく、組み合わせオペレーションを指定する手段はプル・ダウン・メニュー、キーとマウスの組み合わせ、あるいは他の類似の操作を含む。

10 【0029】上述したように幾つかのオペレーションに対して多重選択機能をもっている。多重選択の従来の方法に加えて本発明のメディア拡張管理システムは、テーブルのファイルの同じタイプの多重選択、テーブルの全てのファイルの多重選択、データベースのファイルの同じタイプの多重選択、そしてデータベースの全てのファイルの多重選択のような多重選択機能も持っている。従って、従来の多重選択と上記領域指定多重選択を組み合わせるとユーザは非常に易しく特定オペレーションに対する所望ファイルを選択できる。

20 【0030】ディレクトリ・アイコンを選ぶことによってユーザは選ばれたサブディレクトリに在る全てのファイルに対してオペレーションを行うことができる。このことは一つのサブディレクトリ全体をある場所から別の場所に移し、選ばれたディレクトリの中でファイル間に同じ構造的関係を保つことができることを意味する。

30 【0031】ユーザがテーブル・リストのあるエントリを選択することはそのテーブルの全てのファイルを選択することに等価である。それ故にユーザはそのテーブルの全てのファイルに関してオペレーションを行う事ができる。例えば、“ORG Table”のテーブル・リスト・エントリがオペレーション(即ち、通常の簡単な選択と違ってキーとマウスの組み合わせによって)の対象に選択された場合、このテーブルに属する全てのファイル・アイコンは強調表示される。

【0032】まとめとして、本発明の構成に関して以下の事項を開示する。

40 【0033】(1) コンピュータ・システムにおけるデータベース・テーブルのメディア拡張を管理するユーザ・インタフェースであって、オブジェクトとカーソルとを表示するためのディスプレイを具備し、マウス・ボタンのシングル・クリックにより前記ディスプレイ上に表示されたオブジェクトを選択するべく、又は前記マウス・ボタンのダブル・クリックによって所与のアクションを呼出すべく、もしくは前記マウス・ボタンの押下保持によるオブジェクトの選択と、選択された該オブジェクトの前記ディスプレイ上における移動と、前記マウス・ボタンの解放とからなるドラッグ・アンド・ドロップ・アクションを呼出すべく使用可能なマウスによって前記オブジェクトと前記カーソルとが制御されるコンピュータ・システム上にて実現され、かつ、データベース中の

全てのテーブルのテーブル・リストから構成されるデータ構造を複数の仮想的又は物理的メディアに記憶することにより前記コンピュータ・システム上のデータベース・テーブルのためのメディア拡張を管理するユーザ・インタフェースであって、前記テーブル・リストを更新しかつ複数のメディアの各々に対する階層的ツリー構造のディレクトリ・ルートを作成することによって前記データ・ベースを初期化するステップと、前記複数のメディアの各々についてメディア拡張主ウィンドウ内にルート・ディレクトリ・アイコンを表示するステップと、ユーザによるディレクトリ・アイコン上でのダブル・クリックを検知し、そして前記ディレクトリ・アイコンが、該ディレクトリ・アイコンの下に表示される階層ツリー構造の中に子ディレクトリ・アイコン又はテーブル・パート・アイコンが存在するにも拘わらずこれらを表示していない場合は、表示を拡大することによりこれらの子ディレクトリ・アイコン又はテーブル・パート・アイコンを表示し、一方、存在しない場合は、前記階層ツリー構造の表示を取止めて前記ディレクトリ・アイコンのみを表示するステップと、ユーザによる前記テーブル・パート・アイコン上でのダブル・クリックを検知し、そして前記ユーザに対して、前記テーブル・パート・アイコンに対応するテーブルのフィールドを変更する入力を行うよう促すステップと、ユーザによる一方のディレクトリ・アイコンから他方のディレクトリ・アイコンへの前記テーブル・パート・アイコンのドラッグ・アンド・ドロップ処理を検知し、そしてその結果変更された前記階層ツリー構造の表示中のディレクトリ・パスを更新しかつ前記データベースの前記データ構造を更新するステップと、前記メディア拡張主ウィンドウ中に追加テーブル・パート・アイコンと削除テーブル・パート・アイコンとを表示するステップと、ユーザによる前記追加テーブル・パート・アイコンの前記ディレクトリ・アイコンへのドラッグ・アンド・ドロップ・アクションを検知し、そして前記ユーザに対して、前記ディレクトリ・アイコンに対応するディレクトリへ追加されるべきテーブルのフィールドを変更する入力を行うよう促すステップと、ユーザによるディレクトリ・ツリー中の前記テーブル・パート・アイコンの選択とユーザによる前記削除テーブル・パート・アイコンの選択とを検知し、そして選択された前記テーブル・パート・アイコンを前記メディア拡張主ウィンドウの表示から取除き、かつ前記データベースを更新するステップとを含むユーザ・インタフェース。

(2) 前記メディア拡張主ウィンドウ内に、テーブル・リスト副ウィンドウを表示するステップと、前記メディア拡張主ウィンドウ内に、テーブル詳細リスト副ウィンドウを表示するステップと、ユーザによる前記テーブル・リスト副ウィンドウからの選択を検知し、そしてテーブル・データ構造を構文解析することによって、テーブル詳細リスト・エントリを用いてテーブル詳細リストを

更新するステップと、ユーザによる前記テーブル詳細リスト副ウィンドウからの選択を検知し、そして前記ユーザによる選択がメイン・テーブル・エントリである場合、前記ディレクトリ・ツリー中の全てのテーブル・パートを強調表示し、一方、前記ユーザによる選択がテーブル・パート・エントリである場合、前記ディレクトリ・ツリー中の個々の前記テーブル・パート・アイコンを強調表示するステップとを含む上記(1)に記載のユーザ・インタフェース。

- 10 (3) データベース・テーブルのメディア拡張を管理するコンピュータシステムであって、システム・バスに接続された中央演算処理ユニットと、前記システム・バスに接続され、かつユーザにより制御されるマウスと、前記システム・バスに接続され、かつデータベース中の全てのテーブルのテーブル・リストからなる該データベースのデータ構造を記憶する複数の仮想的又は物理的メディアと、前記システム・バスに接続され、かつマウス・ボタンのシングル・クリックによりディスプレイ上に表示されたオブジェクトを選択するべく、又は前記マウス・ボタンのダブル・クリックによって所与のアクションを呼出すべく、もしくは前記マウス・ボタンの押下保持によるオブジェクトの選択と、選択された該オブジェクトの前記ディスプレイ上における移動と、前記マウス・ボタンの解放とからなるドラッグ・アンド・ドロップ・アクションを呼出すべく前記ユーザにより使用可能な前記マウスによって制御されるグラフィック・ユーザ・インタフェース中のオブジェクトとカーソルとを表示するディスプレイとを有し、前記中央演算処理ユニットが、前記テーブル・リストを更新しかつ複数のメディアの各々に対する階層的ツリー構造のディレクトリ・ルートを作成することによって前記データ・ベースを初期化するべく、前記複数のメディアの各々についてメディア拡張主ウィンドウ内にルート・ディレクトリ・アイコンを表示するべく、ユーザによるディレクトリ・アイコン上でのダブル・クリックを検知して、前記ディレクトリ・アイコンが該ディレクトリ・アイコンの下に表示される階層ツリー構造の中に子ディレクトリ・アイコン又はテーブル・パート・アイコンが存在するにも拘わらずこれらを表示していない場合は、表示を拡大することにより前記子ディレクトリ・アイコン又は前記テーブル・パート・アイコンを表示し、一方、存在しない場合は、前記階層ツリー構造の表示を取止めて前記ディレクトリ・アイコンのみを表示するべく、ユーザによる前記テーブル・パート・アイコン上でのダブル・クリックを検知して、前記ユーザに対して前記テーブル・パート・アイコンに対応するテーブルのフィールドを変更する入力を行うよう促すべく、ユーザによる一方のディレクトリ・アイコンから他方のディレクトリ・アイコンへの前記テーブル・パート・アイコンのドラッグ・アンド・ドロップ処理
- 20
- 30
- 40
- 50

表示中のディレクトリ・パスを更新し、かつ前記データベースの前記データ構造を更新するべく、前記メディア拡張主ウィンドウ中に追加テーブル・パート・アイコンと削除テーブル・パート・アイコンとを表示するべく、ユーザによる前記追加テーブル・パート・アイコンの前記ディレクトリ・アイコンへのドラッグ・アンド・ドロップ・アクションを検知し、そして前記ユーザに対して、前記ディレクトリ・アイコンに対応するディレクトリへ追加されるべきテーブルのフィールドを変更する入力を行うよう促すべく、ユーザによるディレクトリ・ツリー中の前記テーブル・パート・アイコンの選択とユーザによる前記削除テーブル・パート・アイコンの選択とを検知して、選択された前記テーブル・パート・アイコンを前記メディア拡張主ウィンドウの表示から取除き、かつ前記データベースを更新するコンピュータ・システム。

【0034】

【発明の効果】本発明に従うグラフィック・ユーザ・インタフェースを用いると、データベースがハード・ディスクあるいはドライブのパーティションの容量をも越えるものもパーティションの異なるディレクトリに、あるいは異なるメディア(media)のドライブにまたがって分割する機能をもつので、容易にアクセスすることができ、それはこのユーザ・インタフェースは、テーブルに属する全てのファイル(データベースのパート)が素早く、そして一義的に識別されるようにデータベースの全てのテーブル・パートの包括的で地形学的な地図を提供するからである。また、ユーザは作成されたテーブル・パート・エントリを一つのディレクトリ位置から別の位置にドラッグしてドロップできる。新しいディレクトリ・パス(経路)へのタイプ入力は不要である。直接処理手法が広範囲に用いられるので、キーボード入力を少なくして操作を容易にする。

【図面の簡単な説明】

*

*【図1】本発明を実現できるハードウェア構成を示すブロック図である。

【図2】本発明の一つの実現を示すメディア拡張主ウィンドウのディスプレイ図である。

【図3】図2で示される主ウィンドウのディスプレイで使用されるアイコンの例である。

【図4】データベースのテーブル・リストを表すテーブル・リスト副ウィンドウの図である。

【図5】テーブルを選択したときに現れるテーブル詳細副ウィンドウの図である。

【図6】本発明の実施例に従うメディア拡張処理の論理を示すフロー図である。

【図7】本発明の実施例に従うメディア拡張処理の論理を示すフロー図である。

【図8】本発明の実施例に従うメディア拡張処理の論理を示すフロー図である。

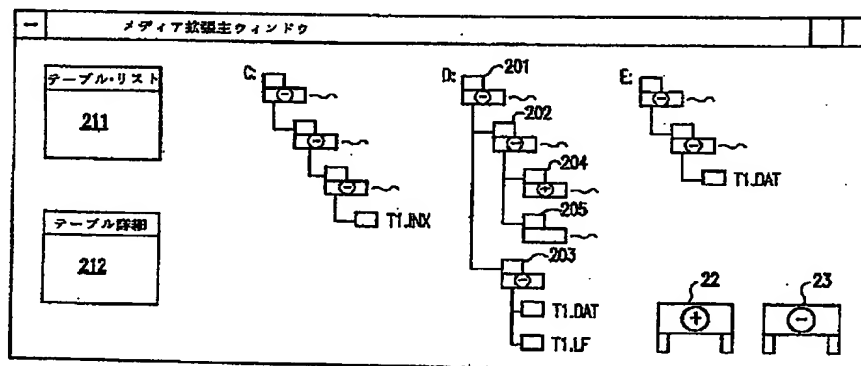
【図9】本発明の実施例に従うメディア拡張処理の論理を示すフロー図である。

【図10】本発明の実施で使用されるテーブル・パートのデータ構造を例示するブロック図である。

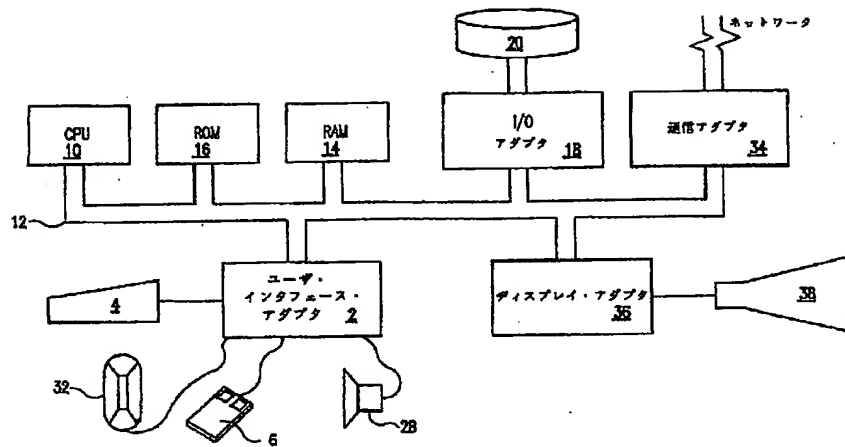
【符合の説明】

- 10 中央処理装置(CPU)
- 14 ランダム・アクセス・メモリ(RAM)
- 16 リード・オンリ・メモリ(ROM)
- 18 入出力アダプタ
- 20 ディスク・ドライブ
- 24 キーボード
- 26 マウス
- 28 スピーカ
- 30 32 マイクロフォン
- 34 通信アダプタ
- 36 表示アダプタ
- 38 ディスプレイ

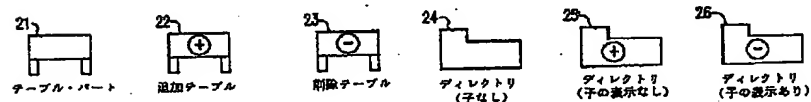
【図2】



【図1】



【図3】



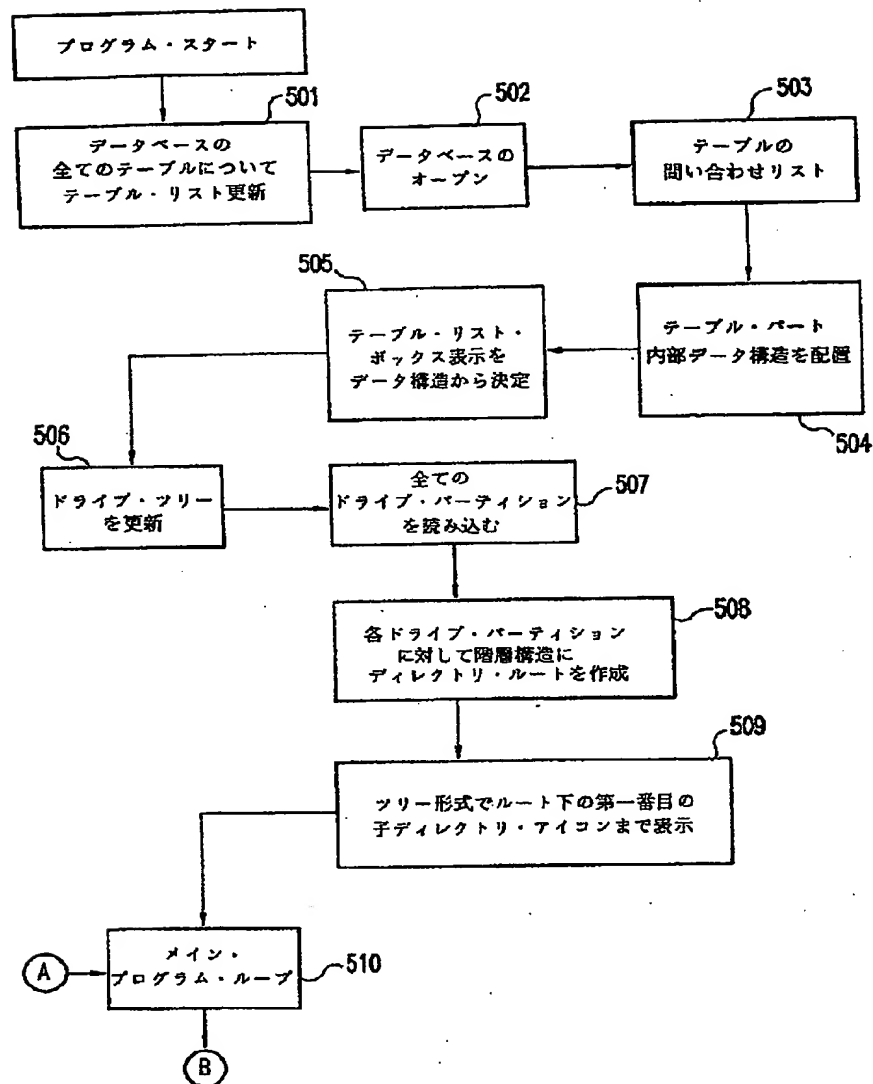
【図4】

—	テーブル・リスト				
ORG	3 FILES	30K	7-3-91		△
T1	4 FILES	53K	10-11-91		
T2	5 FILES	105K	1- 4-91		
STAFF	3 FILES	25K	3-4-91		
⋮					
⋮					▽

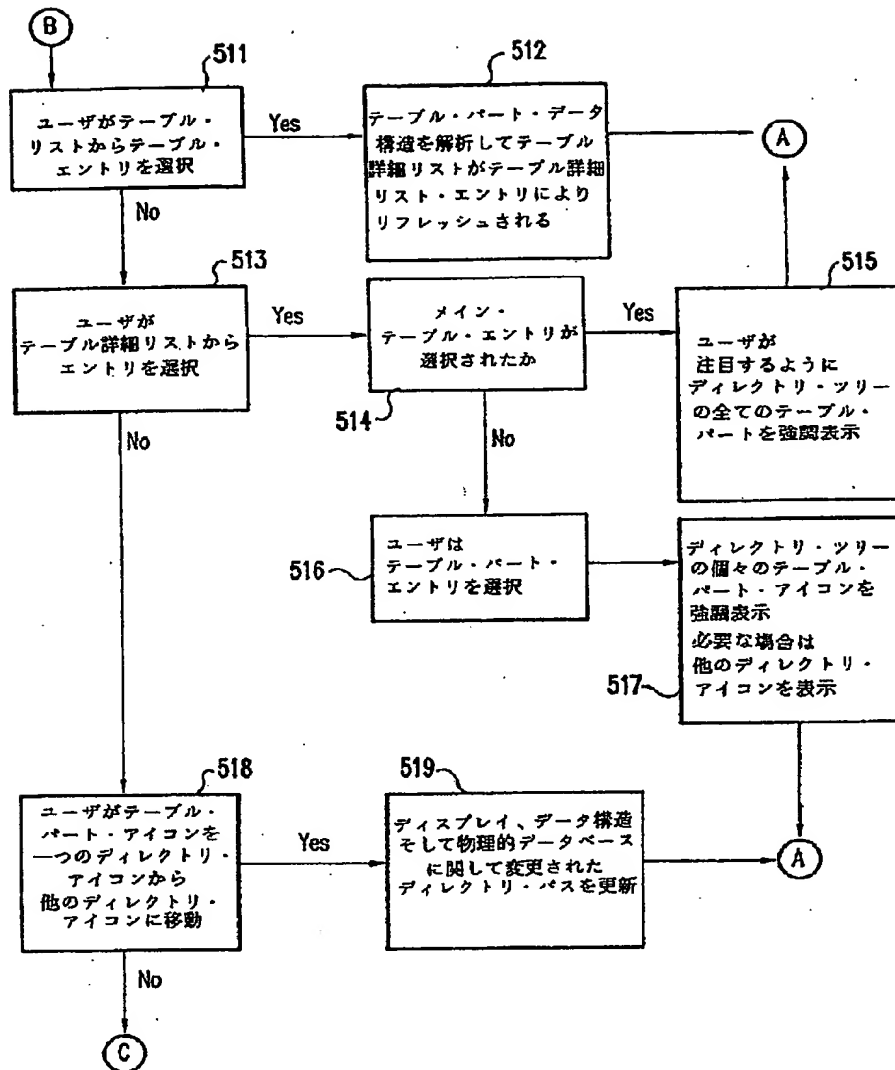
【図5】

—	テーブル詳細			
T1	4 FILES	53K	10-11-91	△
INX	6K	C:\...	8-6-91	
DAT	20K	D:\...	10-11-91	
LF	3K	D:\...	10-9-91	
DAT	24K	E:\...	10-11-91	
T2	...			
⋮				▽

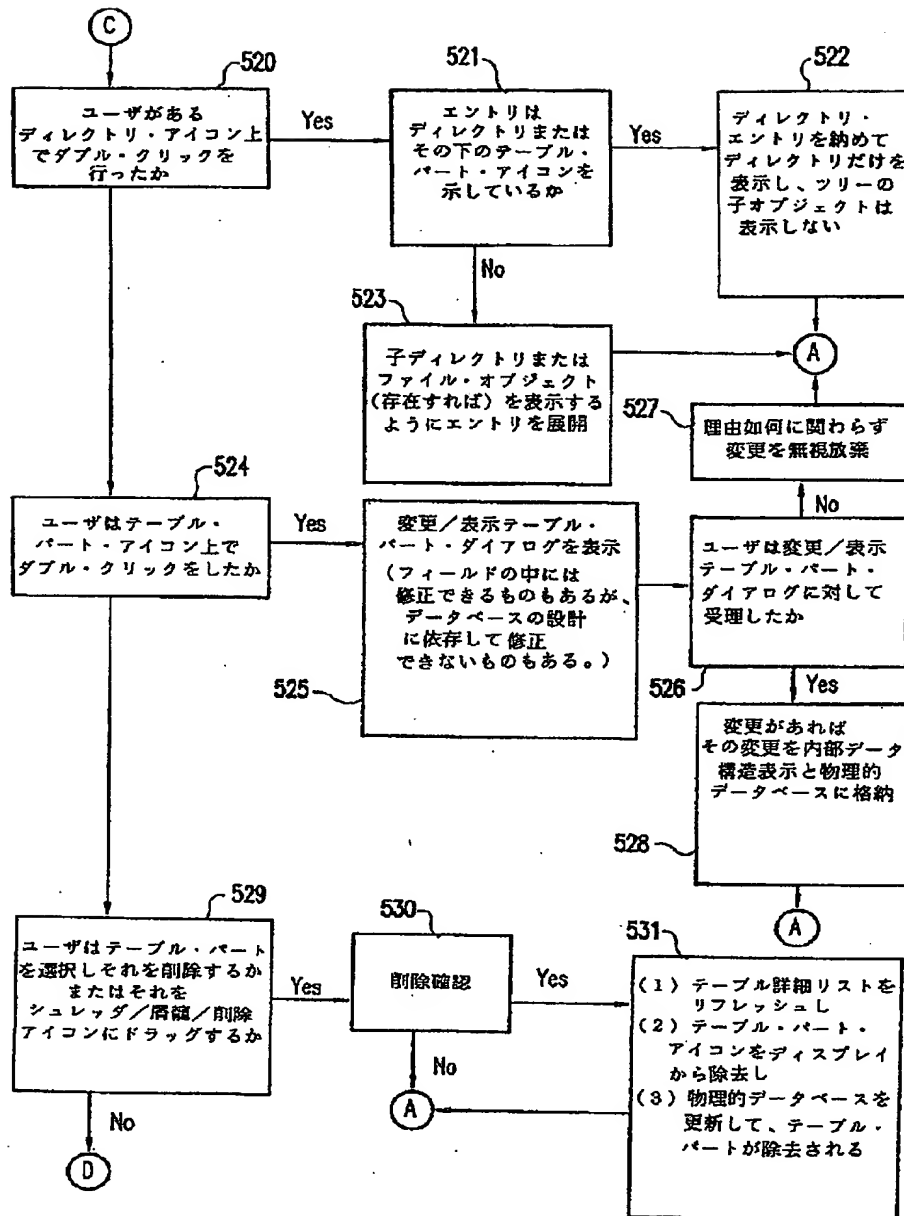
【図6】



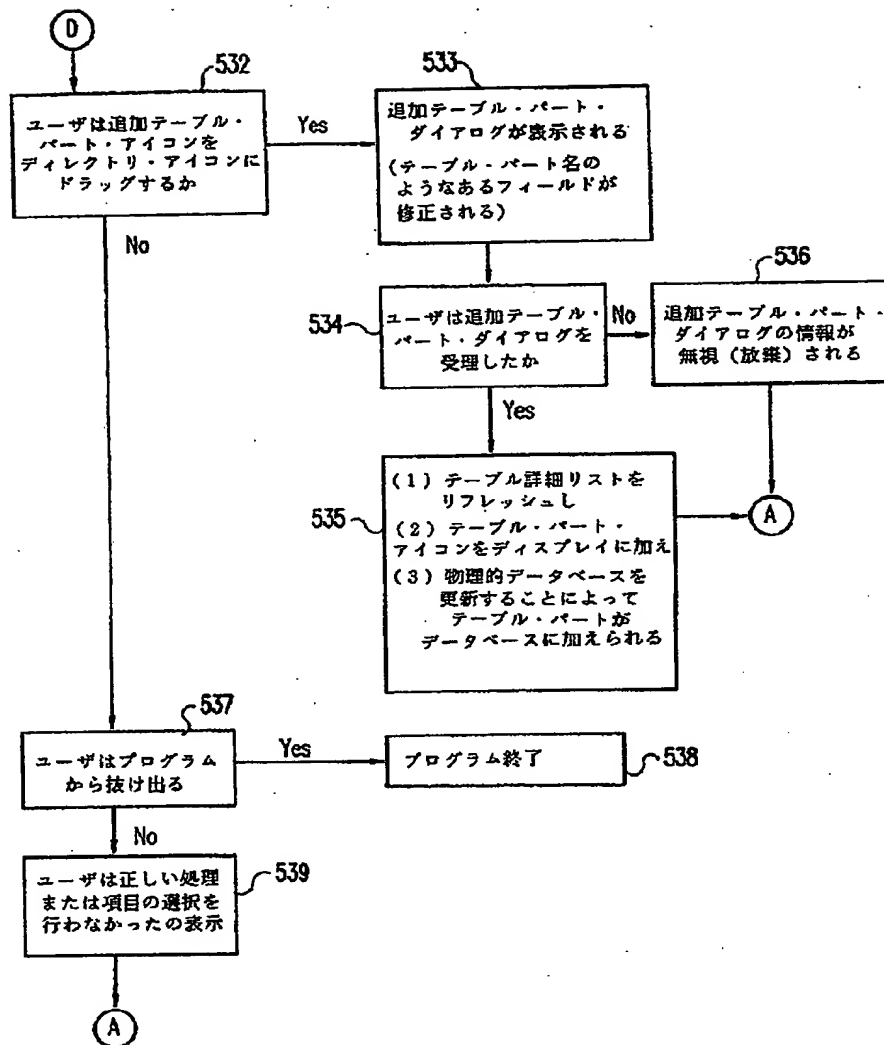
【図7】



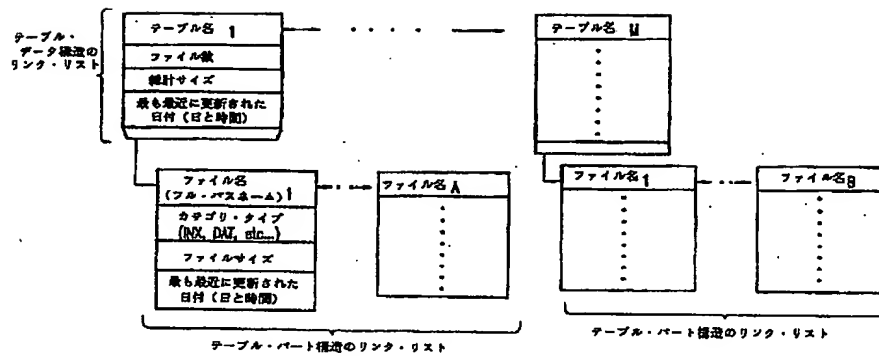
【図8】



【図9】



【図10】



フロントページの続き

(72)発明者 セオドア・ジャック・ロンドン・シュレー
 ダー
 アメリカ合衆国78728 テキサス州、オー
 スチン、ショアライン・ドライブ 3101、
 アpartment・ナンバー1936

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

☐ **BLACK BORDERS**

☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**

☒ **FADED TEXT OR DRAWING**

☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**

☐ **SKEWED/SLANTED IMAGES**

☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**

☐ **GRAY SCALE DOCUMENTS**

☒ **LINES OR MARKS ON ORIGINAL DOCUMENT**

☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**

☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.